

**Техническая архитектура образовательной платформы
LMS Fashion Factory School**

Содержание

1. Назначение документа	3
2. Описание архитектуры LMS Fashion Factory School	3
2.1. Клиентская часть (Frontend)	3
2.2. Серверная часть (Backend)	3
2.3. Базы данных	3
2.4. Облачные хранилища и CDN	3
2.5. API и микросервисы	3
2.6. Системы аутентификации и авторизации	3
2.7. Мониторинг и логирование	4
2.8. Системы резервного копирования и восстановления	4
2.9. Интеграции с внешними сервисами	4
2.10. Контейнеризация и оркестрация	4
2.11. Балансировка нагрузки и масштабирование	4
2.12. Безопасность	4
2.13. DevOps и CI/CD	5

1. Назначение документа

Данный документ в первую очередь ориентирован на команду технических специалистов и архитекторов Заказчика. В документе описывается техническая архитектура образовательной платформы LMS Fashion Factory School.

2. Описание архитектуры LMS Fashion Factory School

2.1. Клиентская часть (Frontend)

- Веб-интерфейс: SPA Веб-приложение разработано с использованием современного фреймворка React, на языках программирования TypeScript, CSS, что обеспечивает высокую интерактивность и отзывчивость интерфейса.

2.2. Серверная часть (Backend)

- Веб-серверы: Ingress для обработки HTTP-запросов и направления их к серверным приложениям.

- Серверы приложений: Основной код платформы, разработан на фреймворке Django, язык программирования Python 3, который обрабатывает бизнес-логику и взаимодействует с базами данных.

2.3. Базы данных

- Реляционные базы данных: PostgreSQL для хранения структурированных данных (пользователи, курсы, транзакции).

- Нереляционные базы данных: Redis для хранения неструктурированных данных и кэширования и выполнения периодических задач

2.4. Облачные хранилища и CDN

- Объектное хранилище: Selectel для хранения файлов, таких как видео уроки и документы.

- CDN: Selectel для быстрой доставки статического контента пользователям по всему миру.

2.5. API и микросервисы

- RESTful API: Используются для взаимодействия между различными компонентами системы.

- Микросервисы: Реализованы для различных функциональных областей, таких как управление пользователями, платежи, аналитика, и развернуты независимо для обеспечения масштабирования в случае изменения нагрузки на отдельные части системы.

2.6. Системы аутентификации и авторизации

- OAuth 2.0 и JWT: Обеспечение безопасного доступа и управления сессиями пользователей.

- SSO (Single Sign-On): Для корпоративных клиентов, которые требуют интеграции с существующими системами аутентификации.

2.7. Мониторинг и логирование

- Мониторинг: Prometheus и Grafana для отслеживания производительности и состояния системы.

- Логирование: EFK Stack (Elasticsearch, Fluentd, Kibana) для сбора, анализа и визуализации логов.

2.8. Системы резервного копирования и восстановления

- Резервное копирование: Регулярные резервные копии баз данных и критических данных с использованием облачных решений.

- Восстановление после сбоев: Планы и инструменты для восстановления данных и работоспособности системы в случае сбоев.

2.9. Интеграции с внешними сервисами

- Платёжные системы: Тинькофф, CloudPayments, и другие локальные платёжные провайдеры для обработки платежей.

- Маркетинговые и аналитические инструменты: Яндекс.Метрика

2.10. Контейнеризация и оркестрация

- Контейнеризация: Docker для создания контейнеров, обеспечивающих изоляцию и переносимость приложений.

- Оркестрация контейнеров: Kubernetes для автоматизации развертывания, масштабирования и управления контейнер изованными приложениями.

2.11. Балансировка нагрузки и масштабирование

- Балансировщики нагрузки: Ingress для распределения входящего трафика между серверами.

- Автоматическое масштабирование: Использование возможностей облачной платформы Kubernetes Managed by Selectel для автоматического увеличения или уменьшения ресурсов в зависимости от нагрузки.

2.12. Безопасность

- SSL/TLS: Шифрование данных для обеспечения безопасной передачи данных между клиентами и серверами.

- Selectel Managed Kubernetes: Гарантирует SLA 99,98%

2.13. DevOps и CI/CD

- Инструменты CI/CD, GitLab, GitLab CI, для автоматизации процессов тестирования и развертывания.

- Управление конфигурацией: Terraform для управления конфигурацией серверов и приложений.